

TOPIC 1: GETTING STARTED

1.1. Installation

> Download Composer

```
composer
```

> Using Composer

```
composer create -project codeigniter4 /appstarter your
appname
```

1.2. Prepare Running Your App

Configure CI4 to display error messages

> Root|env

Change env to .env

> Root|.env

```
# ENVIRONMENT
CI_ENVIRONMENT = development
```

Hosting with Apache

Virtual Hosting

Step 1: In xampp|apache|conf|httpd.conf

```
# Virtual hosts
Include conf/extra/httpd-vhosts.conf
```

Step 2: xampp|apache|conf|extra|httpd-vhosts.conf

```
<VirtualHost *:80>
    ServerName yourappname.local
    DocumentRoot "D:/xampp/htdocs/yourappname/public"
    "
    <Directory "D:/xampp/htdocs/yourappname/public"
    >
        Options Indexes FollowSymLinks
        AllowOverride All
        Require all granted
    </Directory>
</VirtualHost>
```

1.2. Prepare Running Your App (cont)

Step 3: Edit the hosts file

The hosts file is used to map domain names to IP addresses. You can find this file at `C:\Windows\System32\drivers\etc\hosts` (on Windows) or `/etc/hosts` (on Linux or macOS). Add the following line to hosts file:

```
127.0.0.1 yourappname.local
```

Finally

Restart Apache in the XAMPP control panel.

Remove public/index.php/ from URL

Step 1: Change the `App.php` file

Open `projectname/app/Config/App.php`

changes are as follows:

```
public $baseUrl = 'http://localhost:8080';
to
public $baseUrl = 'http://localhost/yourprojectname/';
```

And the second change in the `app.php` file:

```
public $uriProtocol = 'REQUEST_URI';
to
public $uriProtocol = 'PATH_INFO';
```

Step 2: Copy `index.php` and `.htaccess`

Go to `public` directory
Copy `index.php` and `.htaccess`
To
codeigniter app root directory

Step 3: Change the `index.php`

1.2. Prepare Running Your App (cont)

In the root project directory, open `index.php` and edit the following line:

```
$pathsPath = FCPATH . '../app/Config/Paths.php';
```

change TO

```
$pathsPath = FCPATH . 'app/Config/Paths.php';
```

If the above solution is not work for; so you can configure your apache server; as shown below:

In the apache server, the **mode rewrite** is already on. But some default values need to be changed on `/etc/apache2/apache2.conf` file. Following are changes,

First, find

```
<Directory /var/www/>
    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
</Directory>
```

And change `AllowOverride None` to `All` and **save**.

Then enable mode rewrite

Then restart the server,

1.3. Run the application in a browser

Local Development Server

```
php spark serve
```

Virtual Hosting

```
Ex: yourapname.localhost
```

Removing the index.php

```
http://localhost/your_project_name/
```

TOPIC 2: VIEW LAYOUTS

How to integrate an admin layout into CodeIgniter 4 so that it is separate from the public layout.

2.1. Create a folder for admin layout

In the `app/Views` folder,

Create a new folder named `admin` to hold the layout files for the admin page.

For example: `app/Views/admin/layout.php`.

2.2. Create a layout file for the admin page

```
<!--
app/Views/admin/layout.php
In the admin folder, create a generic layout file
for your admin page.
For example: app/Views/admin/layout.php.
In this file, you can define layout elements such
as header, menu, footer,
and content.
For example:
-->
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Admin Layout </title>
</head>
<body>
    <header>
        <h1>Admin Header </h1>
    </header>
    <nav>
        <ul>
            <li><a href="#">Dashboard
shboard</a></li>
```



2.2. Create a layout file for the admin page (cont)

```
> <li><a href="#">Users</a></li>
  <li><a href="#">Settings</a></li>
</ul>
</nav>
<main>
  <?php echo $this->renderSection('content') ?>
</main>
<footer>
  <p>Admin Footer</p>
</footer>
</body>
</html>
```

2.3. Create view files for each admin page

```
app\Views\admin\dashboard.php
<?php $this->extend('admin/layout') ?>

<?php $this->section('content') ?>
  <h2>Dashboard</h2>
  <p> Welcome to the admin dashboard!</p>
<?php $this->endSection() ?>
```

2.4. Create routes for admin page

```
app\Config\Routes.php
$route s-> group( 'ad min',
  ['name space' => 'App\C ont rol ler s\A dmin'],

  functi on( $ro utes) {
    $route s-> get ('d ash board', 'Dashboard::ind
ex');
    $route s-> get ('u sers', 'Users::index');
    $route s-> get ('s ett ings', 'Settings::index
');
  });
```

Here, all routes for the admin page start with the prefix admin, and all the controllers for the admin page are located in the App\Controllers\Admin namespace.

2.5. Create controllers for the admin page

```
app\Controllers\Admin\Dashboard.php
<?php
namespace App\Controllers\Admin;
use CodeIgniter\Controller;

class Dashboard extends Controller
{
  public function index()
  {
    return view('admin/dashboard');
  }
}
```

In each of these controller files, you need to inherit from CodeIgniter\Controller and define a method to display the corresponding page. In this method, you can use the view helper to display the file view corresponding to that page.

TOPIC 1: NAMING CONVENTION

Controller file and class

```
File: UsersController.php
class UsersController extends Controller {}
```

URI Segments

The segments in the URL, in following with the Model-View-Controller approach, usually represent:

```
exam ple.c om / cl ass /m et ho d/ID
```

- ✓ The first segment represents the controller class that should be invoked.
- ✓ The second segment represents the class method that should be called.
- ✓ The third, and any additional segments, represent the ID and any variables that will be passed to the controller.

Work with Subdirectory Controllers

```
namespace
| namespace App\Co ntr oll ers \Fo lde rname;

use
| use App\Co ntr oll ers \Ba seC ont roller;
  Class Classname extends BaseCo ntr oll er()
```

Work with Subdirectory Controllers (cont)

Config/Routes

```
$route s-> get ('/ fol der nam e', 'fo lde rna me/ con trol ler nam e: :me thod');
```

Database Migration (copy)

Creating Databases in the Command Line

```
php spark db:create foo
```

6. Creating a table

Run CLI

```
php spark migrat e:c reate create _na meo ftable
```

Open created file :

- ✓ Add code to up() and down() method

Run CLI

- ✓ migrate: php spark migrate
- ✓ rollback: php spark migrat e:r ollback
- ✓ refresh: php spark migrat e:r efresh
- ✓ status: php spark migrat e:s tatus

Creating and Dropping Tables Document

https://codeigniter.com/user_guide/dbmgmt/forge.html#id4

MySQL Cheat Sheet

<https://cheatography.com/davechild/cheat-sheets/mysql/>

Up() method

```
$fields = [];
$forge ->a ddF iel d(' id');
$forge ->a ddF iel d($ fie lds);
//$for ge- >ad dPr ima ryK ey( 'id');
$forge ->c rea teT abl e(' its core');
```

down() method

```
$forge ->d rop Tab le( 'it sco re');
```

5. Connect to the database

Create database & its user:

Open [PhpMyadmin](#) > Create a database > Create a user

Edit file .env:

```
#-----
# DATABASE
#-----

databa se.d ef aul t.h ostname = localhost
databa se.d ef aul t.d atabase = databa sename
databa se.d ef aul t.u sername = username
databa se.d ef aul t.p assword = password
databa se.d ef aul t.D BDriver = MySQLi
```

Run app in a browser to see if you get any errors. If not, your connection is successful.

Some common CLI commands

```
php spark serve
-> This command starts the development server and serves your CodeIgniter 4 application on http://localhost:8080/.

php spark make:c ontroller MyCont roller
-> This command creates a new controller named MyCont roller in the app/Co ntr ollers directory.

php spark make:model MyModel
-> This command creates a new model named MyModel in the app/Models directory.

php spark make:m igr ation create _us ers _table
-> This command creates a new migration file for creating a users table in your database.

php spark migrate
-> This command runs any pending database migrat - ions.

php spark db:seed MySeeder
-> This command seeds your database with data using the specified seeder class name.

/These are just a few examples of the many CLI commands available in CodeIgniter 4. You can run php spark to see a list of all available commands and their descriptions./
```

